USF UNIVERSITY OF SAN FRANCISCO

CHANGE THE WORLD FROM HERE

Assertions

Debugging Strategies

Chris Bryan ◆ cbryan2@usfca.edu
Department of Computer Science

Introduction

- Makes programming assumptions (and hopes?) explicit
  - Assuming input in private method is never null
- Can embed throughout code
  - No output produced if assertion is correct
- Can enable/disable assertions via compiler flag
  - Add –ea flag as vm argument when running

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Using Assertions

- Remember, assertions can be disabled!
- Do not use for input validation in public methods
- Do not use assertions to perform actions necessary for functionality of program

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Use for Internal Invariants

```
if (i % 3 == 0) {
  ...
} else if (i % 3 == 1) {
  ...
} else {
  assert i % 3 == 2;
  ...
}
```

```
switch (booleanBit) {
case 0:
  ...
case 1:
  ...
default:
  assert false;
}
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Use for Control Invariants

```
void firstFound(List<String> elements, String find) {
  for (String element : elements) {
    if (element.equals(find)) {
       System.out.println(“Found: ” + element);
       return;
    }
  }
  // we know element is in list, so should should never reach this point
  assert false;
}
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Use for Private Preconditions

```
public void addWord(String word) {
    if (word != null) {
        words.add(convertString(word));
    }
}

private String convertString(String text) {
    assert text != null;
    ...
    return text;
}
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Use for Postconditions

```
public long copy(Path source, Path destination) {
  try {
      copyFile(source, destination);
  } catch (Exception ex) {
      return -1;
  }
  assert Files.size(source) == Files.size(destination);
  return Files.size(destination);
}
```

http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Summary

- Do use to indicate any assumptions made by programmer
- Do use to verify input in private methods
- Do NOT use to validate user input in public methods
- Do NOT use to perform necessary operations

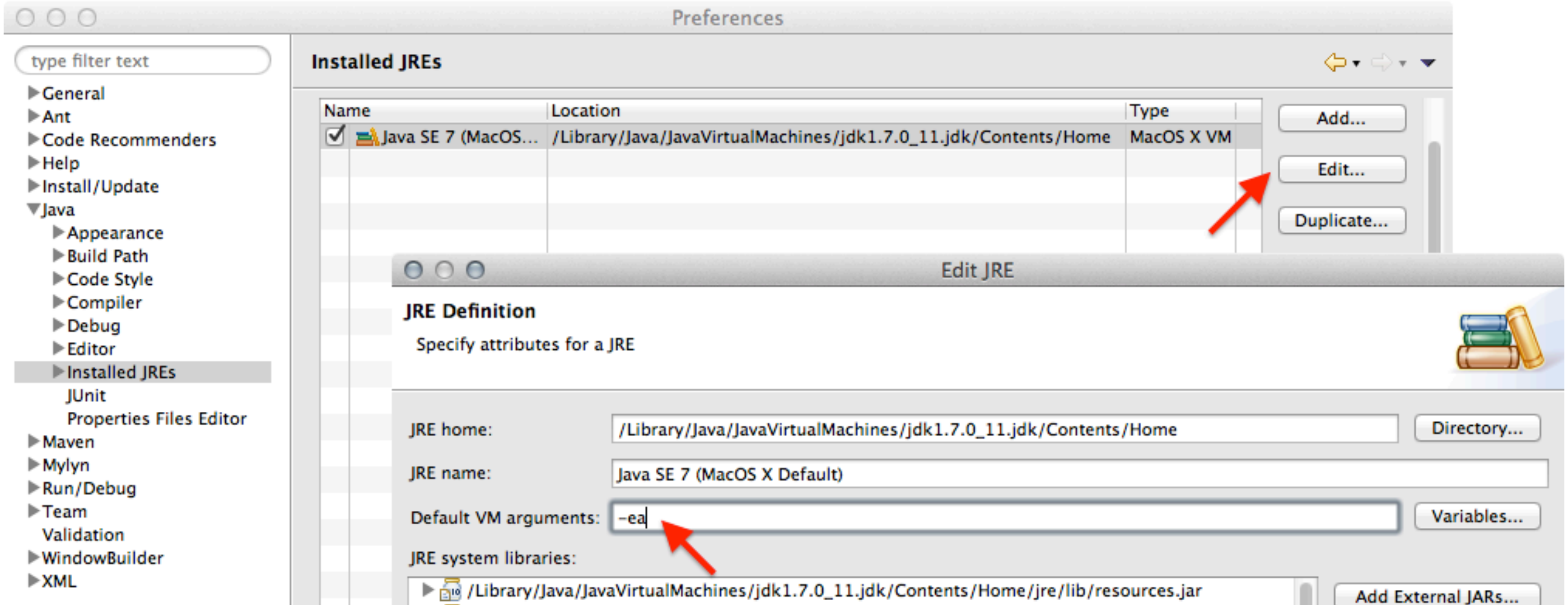
http://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html

Enabling Assertions

- Assertions are not enabled by default
  - Require –ea (enable assertions) flag when running
- Two methods for enabling in Eclipse
  - Always enable assertions for all Java projects
  - Only enable assertions for certain Java Projects

http://tutoringcenter.cs.usfca.edu/resources/enabling-assertions-in-eclipse.html

Always Enable Assertions



Preferences

type filter text

- General
- Ant
- Code Recommenders
- Help
- Install/Update
- Java
  - Appearance
  - Build Path
  - Code Style
  - Compiler
  - Debug
  - Editor
  - Installed JREs
  - JUnit
  - Properties Files Editor
- Maven
- Mylyn
- Run/Debug
- Team
- Validation
- WindowBuilder
- XML

Installed JREs

Name	Location	Type
☑ Java SE 7 (MacOS...	/Library/Java/JavaVirtualMachines/jdk1.7.0_11.jdk/Contents/Home	MacOS X VM

Edit JRE

JRE Definition

Specify attributes for a JRE

JRE home: /Library/Java/JavaVirtualMachines/jdk1.7.0_11.jdk/Contents/Home

JRE name: Java SE 7 (MacOS X Default)

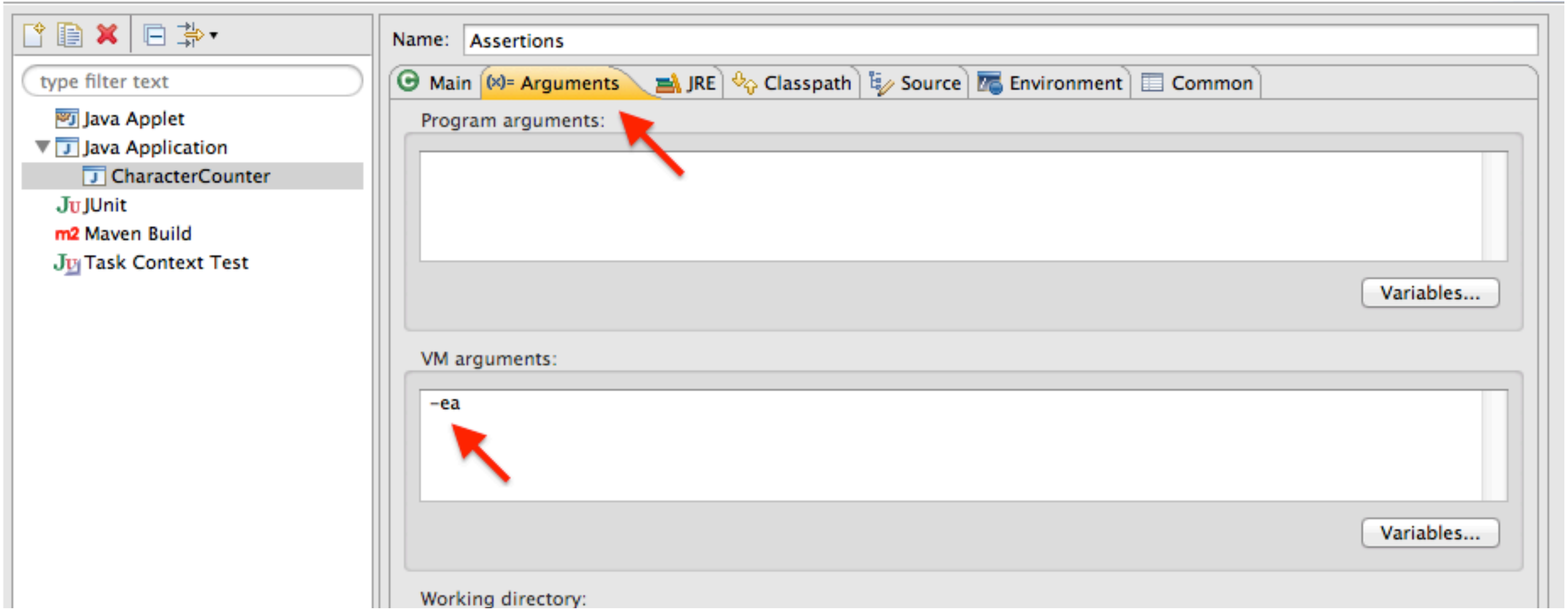
Default VM arguments: -ea

JRE system libraries:

/Library/Java/JavaVirtualMachines/jdk1.7.0_11.jdk/Contents/Home/jre/lib/resources.jar

Preferences » Installed JREs » Java SE 8 » Edit… » Default VM arguments » -ea » Finish

Enable Pre-Project Assertions



Name: Assertions

Program arguments:

VM arguments:

```
-ea
```

Run Configurations » Arguments » VM arguments » -ea » Run

USF UNIVERSITY OF SAN FRANCISCO

CHANGE THE WORLD FROM HERE